

ARMY RESEARCH LABORATORY



# Creation of a New Modeline for a Nonstandard, Unsupported Display Size in a Red Hat Linux, XFree86 Environment

Richard D. Gopaul

ARL-TR-853

April 2001

Approved for public release; distribution unlimited.

20010705 015

The findings in this report are not to be construed as an official Department of the Army position unless so designated by other authorized documents.

Citation of manufacturer's or trade names does not constitute an official endorsement or approval of the use thereof.

Destroy this report when it is no longer needed. Do not return it to the originator.

# Army Research Laboratory

Adelphi, MD 20783-1197

---

ARL-TR-853

April 2001

## Creation of a New Modeline for a Nonstandard, Unsupported Display Size in a Red Hat Linux, XFree86 Environment

Richard D. Gopaul

Computational and Information Sciences Directorate

---

## Abstract

---

Nonstandard display modes for irregular display sizes are not readily supported by the Red Hat Linux operating system. This lack of support leads to the wasting of screen space when one uses either the text mode or X Windows interfaces. The problem becomes especially prevalent in the X environment. The system will not correctly detect the size of the screen and may project images off screen or images that are too small to take advantage of the full viewable capacity of the screen. To correct this problem, a new modeline must be created that defines the total viewable area of the monitor. This newly created modeline will allow the X server to display any and all images within the bounds of the screen. To accomplish this task, one must modify the XF86Config file to include the newly created modeline, as well as, any display options necessary to the display adapter for proper image display. This report focuses specifically on the creation of a nonstandard "800 × 480" modeline for the Toshiba Libretto 100CT. This report should serve as an adequate guide for the creation of other nonstandard modelines.

---

## Contents

---

1. Introduction	1
2. Supplemental Information	2
3. Example 800 × 480 Modeline	6
4. Conclusion	14
5. Acknowledgments	15
Appendix A. Example of a Working XF86Config file generated for the Toshiba Libretto 100CT	17
Distribution	21
Report Documentation Page	23

## Tables

1. XF86Config file format .....	2
2. Listing of standard and accelerated X servers .....	4

---

## 1. Introduction

---

The majority of monitors manufactured today are designed according to standard display sizes. These display sizes are readily supported by most operating systems and so can be exploited to their full viewable capacity. Even if the monitor itself is not directly supported by means of a manufacturer specific driver, a generic driver based on a standard display mode can usually be used. The same logic unfortunately applies to a nonstandard display size. When a nonstandard display size is encountered, just as with the standard display size, the computer system will attempt to use a predefined standard display mode. However, in this example, the defined display mode is imposed onto a nonstandard display size (e.g., a  $800 \times 600$  display mode imposed on a  $800 \times 480$  screen). This mismatch can lead to many display issues such as a display area too large for the screen, resulting in images that are not visible, or a display area smaller than the screen, ultimately wasting valuable screen space. One can perform a search to locate a driver or mode specific to the monitor; however, the likelihood of finding such a driver for a nonstandard mode, especially in a Linux environment, is remote since most Linux drivers are written and contributed by private individuals on an as needed basis.

Depending on the operating system, modifying current display modes or creating a new display mode (modeline) for the system may or may not be possible. This report demonstrates the creation of a new modeline in a Linux XFree86 display environment. In this environment, one can modify display settings and create new modelines specific to the viewable area of the monitor. These actions maximize the use of the monitor so that all available space is used and all projections appear within the bounds of the monitor. In essence, a custom display mode configuration (modeline) will be generated. Before taking effect, this modeline will need to be inserted into certain sections of the XF86Config file. It is advisable to read any documentation on the display adapter specific to the computer to determine if any supplemental hardware display options are necessary.

The descriptions and examples provided in the following sections present the creation of a modeline for a  $800 \times 480$  display size. This new modeline is generated for a Toshiba Libretto 100CT, with Linux as the operating system and XFree86 3.3.6-20 as the display environment. The Toshiba Libretto 100CT is manufactured with an  $800 \times 480$  LCD screen size and comes preloaded with a Windows operating system. Drivers for this nonstandard screen size are readily available for the Microsoft Windows environment but are virtually nonexistent for any other operating system. The newly created modeline is used to properly configure the XF86Config file for use with the Linux SVGAX server. An explanation of certain terms used is presented in section 2.

---

## 2. Supplemental Information

---

XFree86 system (X) forms the basis of the graphical user interface for Red Hat Linux. It consists of multiple X servers, each of which is designed to support a specific set of graphics cards. XFree86 is included in most recent distributions of Linux and only needs to be configured properly to take advantage of its features. The `/etc/X11/XF86Config` file contains all of the vital information necessary to begin an X server session. Upon execution, the selected X server will read from this XF86Config file and use the valid information to configure necessary parameters. Without correct (or at least generic) values contained within this file, the X server will not run properly or will crash. The format for the XF86Config file is shown in table 1.

Linux offers various tools with which to configure XFree86. These tools, for the most part, only modify the XF86Config file in an attempt to generate correct configuration options necessary for the selected X server to function. The most common configuration tools include (but are not limited to) (1) "Xconfigurator," (2) "XF86Setup," (3) "xf86config," and (4) "xvidtune." They all should produce the same end results; however, they each possess unique qualities that may satisfy a certain need at a given time:

1. Xconfigurator is a graphical configuration tool for XFree86 that is included with the Red Hat Linux distribution. Upon execution, Xconfigurator will generate a new default XF86Config file in which it will include modelines for most standard, supported display modes. It will then manipulate the XF86Config file, inserting proper values,

Table 1. XF86Config file format.

Section	Description
Files	Tells the X server where colors, fonts, or specific software modules are located.
Module	Tells the X server what special modules should be loaded.
ServerFlags	Flags that allow or deny special actions, such as core dumps, keyboard server shutdown, video-mode switching, etc.
Keyboard	Tells the X server what type of keyboard to expect and what settings to use.
Pointer	Tells the X server what type of mouse to expect and how to handle the buttons.
Xinput	A special section for devices, such as graphics pads or styluses.
Monitor	Specific details and settings for the monitor, such as name, horizontal sync, vertical sync ranges, and modelines.
Device	Information about the video chipset, such as video RAM or Clockchips.
Screen	Information about which X server to use, color depth, and screen size.

and will create any symbolic links to the selected X server as necessary. The Xconfigurator, however, will allow an incorrect or incomplete XF86Config file to be saved, which may crash the X server during initialization. This capability may or may not be useful or harmful, but in our case it does prove useful as long as any needed information is backed up beforehand.

2. XF86Setup is another graphical configuration tool that is included with XFree86. It, however, will start the VGA16 server in a low-level  $640 \times 480$  display mode before accepting any configuration options. Like the Xconfigurator, XF86Setup will also create and modify the XF86Config file and create any necessary symbolic links to the X server. Unlike the Xconfigurator though, XF86Setup will neither save an invalid configuration nor include extra modelines or options into the XF86Config file for standard display modes. XF86Setup will only create a new XF86Config file when the X server is able to start (using the options selected during configuration) and will include only modelines for the display modes selected during configuration.
3. The xf86config tool can also be used to create and modify the XF86Config file and generate any necessary symbolic links to the X server. Unlike either the Xconfigurator or XF86Setup tools, XF86Config is a text-based configuration tool. It prompts the user for configuration information and uses the entered information to generate a new XF86Config file. Like the Xconfigurator, XF86Config will also include extra modelines and options for standard display modes. Additionally, it will prompt the user toward the end of the configuration whether or not to enable a virtual screen size, something not done by either of the first two tools.
4. The xvidtune tool is a "client interface to the XFree86 X server video mode extension (Xfree86-VidModeExtension)."<sup>1</sup> It is included with any recent version of XFree86. By default, xvidtune will contain the settings for the currently running modeline, and when invoked within the X-windows environment, xvidtune will appear as a window with various buttons and scroll bars that allow for the modification of the currently running display mode; however, any modifications made will not affect the system directly. The xvidtune utility serves only to test different display modes. It will not modify the current XF86Config file but instead will resize and re-position the current display and/or printout, if requested, the current modeline in proper format within a terminal window. xvidtune is not used in this example.

---

<sup>1</sup>K. Keithley, X Consortium, xvidtune(1) [man page] [/var/catman/X11R6/cat1/xvidtune.1x.gz](#).



XFree86 consists of various X servers, most of which are specific to a certain type of graphics device. These servers are located in the "/usr/X11R6/bin/" directory (see table 2).

Just as the tools for the configuration of the XF86Config file are unique, the different X servers are also distinct in the way that they interpret the XF86Config file. They each look for specific value ranges, without which they will not function. The X server chosen in the XF86Config file determines the server that is invoked upon implementation of X. Also the "/etc/X11/X" link must point to the server selected within the XF86Config file or an error message will be returned. Usually the X configuration tools will establish this link.

Standard display modes supported by Linux are—

- 640 × 480,
- 800 × 600,
- 1024 × 768,
- 1280 × 1024,
- 1152 × 864, and
- 1600 × 1200.

Modelines for these display modes, as well as other more infrequently used display modes, are defined by the Xconfigurator and XF86Config tools. The correct format for a modeline (for example, 800 × 600) is as follows:

Modename	Clock	Horizontal timing				Vertical timing			
800 × 600	36	800	824	896	1024	600	601	603	625

The Modename section is exactly what it appears to be, an identifier for the modeline. The X servers will use this identifier when choosing between modes. The clock section refers to the driving clock frequency of the video card (the total number of pixels per second that it can write to the screen). The horizontal and vertical timing sections consist of four separate numbers, each representing the number of horizontal/vertical scans performed per second. The values that appear in the horizontal timing section, however, must be divisible by 8. These numbers (beginning at the horizontal

Table 2. Listing of standard and accelerated X servers.

Server	Description
XF86_Mono / XF86_VGA2	Monochrome X server
XF86_VGA16	4-bit, 16-color X server
XF86_SVGA	8/16/24/32-bit, 256-color X server
XF86_S3, Mach32, Mach8, 8514, P9000, AGX, W32, Mach64, I128, and S3v	Accelerated X servers

timing column of the example modeline, in order from left to right) represent<sup>2</sup>—

1. The number of visible lines that will appear on the display.
2. The line number at which the horizontal sync pulse will begin.
3. The line number at which the horizontal sync pulse will end.
4. The total horizontal frame length.

The same format applies to the vertical timing section of the example modeline as well. These numbers (beginning at the vertical timing column of the modeline, in order from left to right) represent—

1. The number of visible lines that will appear on the display.
2. The line number at which the vertical sync pulse will begin.
3. The line number at which the vertical sync pulse will end.
4. The total vertical frame length.

Most of these values can usually be obtained from the user's manual provided with the video card and monitor; however, generic values such as the ones provided by the Xconfigurator and XF86Config tools will usually suffice. An example of the creation of a 800 × 480" modeline is presented in section 3.

---

<sup>2</sup> E. Raymond, XFree86 Video Timings HOWTO. [/usr/doc/HOWTO/Xfree86-Video-Timings-HOWTO](#), ver. 4.2 (28 January 2000).

---

### 3. Example 800 × 480 Modeline

---

The following example depicts the procedures to create a 800 × 480 modeline. This example documents the creation of a 800 × 480 modeline, the maximum display capacity for a Toshiba Libretto 100CT.

#### I. *Presetup*

1. Backup a copy of the current “/etc/X11/XF86Config” file, if one exists. This is always a good practice.
2. Consult the manuals provided with your hardware and document—
  - The make and model of the video card.
  - The amount of video memory available to the video card.
  - The maximum clock frequency of the video card.
  - The maximum resolution of the monitor.
  - The maximum horizontal and vertical frame lengths of the monitor.
  - The maximum refresh rate of the monitor.

If some or all of this information is not provided, one of the X configuration utilities can most likely probe for it.

In our example, not much information is provided about the Toshiba Libretto 100CT besides the make and model of the video card, the video memory, and the size and type of the screen:

Make: NeoMagic

Model: NM2160 (MagicGraph 128XD)

Video memory: 2 MB

Screen type: LCD Panel

Screen size: 800 × 480

Even with this limited information, it is still possible to derive a working XF86Config file:

#### II. *Setup*

1. Creation of general X86 Config file.
  - (a) Upon execution of the Xconfigurator, it will perform a probe to determine the type of video card the system contains. It should return a value equivalent to the information just mentioned: “Neomagic Corporation | NM2160 (MagicGraph 128XD)” with a selected X server

of "SVGA." Thus far, this information is correct and "Ok" should be selected from the dialogue box to continue the setup.

- (b) Next, Xconfigurator will attempt to determine the type and size of monitor present. It will display a list of known monitors and resolutions from which to choose. Since the libretto's onboard screen is known to be an LCD panel monitor, the "l" key on the keyboard should be pressed until the LCD panel portion of the list appears. This list contains three possible choices:

- LCD Panel 1024 × 768,
- LCD Panel 640 × 480, and
- LCD Panel 800 × 600.

Our monitor is none of these sizes but instead a combination of two of them (800 × 600 and 640 × 480) that will produce the correct size of 800 × 480. To begin, we select the "LCD Panel 640 × 480" monitor and choose "Ok" from the prompts below to continue.

- (c) Xconfigurator will then need to set up the default resolution and color depth. It can do this one of two ways: by either attempting to probe the system for the correct values or having the information supplied by the user. Choose the "Don't Probe" option here, because the probe often cannot correctly determine the amount of video memory. Misdetection of the video memory will in turn have an effect on the color depth, resulting in a lower value than can actually be managed by the system.
- (d) Xconfigurator will then prompt for the amount of video memory contained in the system. Since this value is known to be 2 MB, scroll down the list using the "↓" key and select the "2 MB" value. We then select "Ok" to continue the setup. To move from the scroll list of video memory values to the "Ok" button, press the "Tab" key.
- (e) Next, Xconfigurator will prompt for the type of Clockchip contained in the system. Since this value is not known, select the "No Clockchip Setting (recommended)" option. Select "Ok" from the bottom two menu choices to continue.
- (f) The next menu will present a list of possible video modes based on the previous set of options entered. From this list, choose the option for 640 × 480 under the "16 bit" section. To accomplish this, use the "Tab" key to move between sections, and when the desired value is reached, use the "Space Bar" to make the selection. Next, use the "Tab" key again to select the "Ok" button from the bottom of the menu.
- (g) Finally, the Xconfigurator will attempt to start the SVGA X server with the provided information. Select the "Ok" button from the dialogue. X will then start; however, the bottom half of the desktop will not be visible on the screen. This error was found to be a problem

with the default hardware settings used by the video card. This is corrected later in the configuration.

After X initiates, it will query the user to ensure that the display settings are functioning properly and that the images are being displayed as they should be. Select the "Yes" option from the dialogue. Another dialogue will then appear that will ask if X should initiate whenever the system boots up. Select the "No" option from this dialogue especially since, as it stands, part of the desktop is not visible in X. Finally, a dialogue will appear that will inform the user that the XF86Config file has been written based on all of the provided information. Select the "OK" option from the bottom of the dialogue to end the X session. At this point, Xconfigurator has terminated and has written the new XF86Config file.

Note: Upon the initiation of the "startx" command, the SVGA server should successfully start; however, the image appears to be too large to fit on the screen. The entire bottom portion of the screen is not visible. This problem is later found to be a result of the default video card settings. By default, the video card will stretch the image if it is not large enough to fill on the screen. In doing this, it attempts to stretch the  $640 \times 480$  display into a screen size of  $800 \times 600$ . Since the display adapter does not recognize the screen as having dimensions of  $800 \times 480$  pixels, but instead  $800 \times 600$  pixels, it attempts to stretch the image to the size of  $800 \times 600$ , which in turn results in the loss of the bottom portion of the picture.

2. Creation of an XF86Config file tailored specifically for the libretto. The following procedures will create a cleaner XF86Config file, containing only required information that can be easily modified later.
  - (a) Save the current XF86Config file under a new name (in this example the current XF86Config file is saved as XF86Config.save).
  - (b) Run the XF86Setup configuration tool by typing "XF86Setup" at the command prompt. Press the "Enter" key when prompted. The VGA16 server should automatically start and display the XF86Setup menus in a  $640 \times 480$  mode.
  - (c) Individually select each menu from the tabs shown at the top of the screen, starting with the "Mouse" section. Upon clicking on the "Mouse" tab, a dialogue will appear listing possible options available to configure the mouse. Click on the "Dismiss" option at the bottom of the dialogue to close it. Now, from within the mouse menu, the proper mouse settings should be selected. In this example, select "ps/2" as the type of mouse from the top set of options. Under the

"Mouse Device" menu, `"/dev/psaux"` should be selected to indicate that the mouse is attached to the ps/2 port. Finally, since the Libretto 100CT only comes with two mouse buttons, the "Emulate3Buttons" option should be selected from below the "Mouse Device" menu. All other options need not be modified.

- (d) Next select the "Keyboard" tab from the top of the screen. All of the default settings should be adequate for the keyboard of the Libretto. The "Model" option should be set to "Generic 101-key PC" and the "Layout" (language) option should contain the value "U.S. English." The "Variant" option should be blank. All other menus on that page should retain the default settings.
- (e) Select the "Card" tab from the top of the screen.
  - Scroll down the list of video cards until the "NeoMagic (laptop notebook)" entry is found. Use the mouse to highlight it.
  - Click on the "Detailed Setup" button and a new set of options should appear. From this menu, select the "SVGA" server under the "Server" section. The "Chipset," "RamDac," "ClockChip," and "Selected options" sections should all be left blank. Under the "Video RAM" section, choose the "2Meg" button for the amount of video memory.
- (f) Select the "Monitor" tab from the top of the screen. From the list of monitor types, select the "Super VGA, 800 × 600 @ 56 Hz" option.
- (g) Select the "Modeselection" tab from the top of the screen.
  - From the list of modes, select the "640 × 480" option.
  - From the bottom of the menu, select the "24bpp" color-depth option.
- (h) Click the "Done" button from the bottom of the screen.
  - XF86Setup will start the SVGA server using the options specified. X should temporarily start, then a menu will appear asking whether or not to save the XF86Config file or to run the xvidtune application.
  - Choose the "Save the configuration and exit" option.
  - Another screen should appear describing if the new XF86Config file could be written or not. If it was successfully saved, choose "Okay" to exit.

There should now be two XF86Config files saved in the `"/etc/X11"` directory: the backup file (XF86Config.save) created using Xconfigurator and the "XF86Config" file just created using XF86Setup. The purpose of these two files will be to construct one final XF86Config file that will properly implement the 800 × 480 modeline. Information will be extracted from the

previously saved XF86Config file (XF86Config.save) and will be used in the creation of the 800 × 480 modeline.

### III. Final Configuration

In this step, the backup XF86Config file generated (Presetup) using the Xconfigurator will only be referred to as "XF86Config.save."

The "Monitor" section of both the newly created XF86Config file (in Setup) and the XF86Config.save file should, by default, contain VertRefresh (vertical refresh rate) values of 60 Hz. The "monitor" section is also where the modelines used by the X server are stored. We will use this default refresh rate to choose a corresponding 800 × 600 modeline from the XF86Config.save file.

- (1) Open the XF86Config.save file (either by typing cat, more, etc).
  - Scroll down to the "Monitor" section of the file, skipping over all other information.
  - Default modelines generated by the system should appear near the end of the "Monitor" section.
  - Locate a 800 × 600 modeline with a comment such as, 800 × 600 @ 60 Hz, 37.8 kHz hsync" This comment indicates an 800 × 600 modeline with a vertical refresh rate of 60 Hz.
  - Record all the modeline information below the comment, including the comment, i.e., 800 × 600 @ 60 Hz, 37.8 kHz hsync Modeline "800 × 600" 40 800 840 968 1056 600 601 605 628
  - Exit the XF86Config.save file.

We now have values for the "800" portion of our 800 × 480 modeline. The values for the "480" portion of the modeline are already contained in the XF86Config file in the 640 × 480 modeline.

2. Open the XF86Config file with a text editor, such as Vi, Pico, etc.
  - Scroll down to the "Monitor" section.
  - Locate the "Modeline" entry (XF86Setup should only have defined one modeline). It should currently read, "Modeline 640 × 480 25.18 640 664 760 800 480 491 493 525."
  - Replace the horizontal values "640 664 760 800" from the current modeline with the values "800 840 968 1056" that were obtained from the XF86Config.save file and also replace the "640" from the 640 × 480 identifier with "800." The modeline should now read, "Modeline "800 × 480" 25.18 800 840 968 1056 480 491 493 525."
3. Scroll down the XF86Config file to the "Screen" sections. There should be several of them, one for each X server. Locate the "Screen" section that corresponds with the SVGA server (Driver "SVGA"). This is the only "Screen" section that is really needed in the XF86Config file. All

the other "Screen" sections pertain to other X servers, such as the monochrome server or the accelerated servers. These superfluous sections can be deleted later to minimize the size and complexity of the file. For our purposes, we only need to work with the SVGA server.

- From within the SVGA Screen section, modify all of the  $640 \times 480$  values to read  $800 \times 480$  instead.
  - Save the XF86Config file and exit.
4. Attempt to "startx."
- More than likely, the X server will crash and warnings will appear on the screen.
  - Issue the command "startx >& log." This will create a log file with all the messages and warnings displayed during the X server initialization and crash.

In reading through this file you will find lines that are preceded with "(\*\*)" and "(--)." The "(\*\*)" indicates that the server was successful in configuring that option or mode, while the "(--)" indicates an error occurred during the configuration and/or initialization of an option or mode. There are also lines that blatantly state "Warning:." Starting from the first "(--)," a message should be displayed that "SVGA: Mode  $800 \times 480$ " needs hsync freq of 23.84 kHz. Deleted." This message means that the modeline that we have provided requires a hsync frequency of 23.84 kHz. The "HorizSync" value-range from the "Monitor" section of the XF86Config file will need to be modified to include a hsync frequency of 23.84 kHz. The three following warnings pertain to the fonts used by the X server. According to the system, the `/usr/X11R6/lib/X11/fonts/local,` and `/usr/X11R6/lib/X11/fonts/100-dpi` directories do not exist. This means that the 100-dpi fonts are not installed, so the system will use the 75-dpi fonts instead. The lines that reference the 100-dpi fonts need only to be deleted from the XF86Config file to resolve this issue. The other errors below the font errors are merely complaints from the SVGA server that it cannot find a valid  $800 \times 480$  mode from which to initialize itself (because it deleted it as a result of the above errors).

5. To correct for these problems, open the XF86Config file for editing once again.
- First, scroll down to the "Files" section and delete the entries for the missing fonts.  
`FontPath "/usr/X11R6/lib/X11/fonts/local"`  
`FontPath "/usr/X11R6/lib/X11/fonts/100dpi:unscaled"`  
`FontPath "/usr/X11R6/lib/X11/fonts/100dpi"`
  - Next, scroll down to the "Monitor" section and modify the "HorizSync" range value. Since the SVGA server needs a hsync



frequency value of 23.84, modify the "HorizSync" range value to include this value. Set the lower limit of the range to a value such as "23.5" so that the new "HorizSync" range now reads "23.5-35.1." This new range should not damage the LCD. Also, the "HorizSync" value can be set to the exact value needed by the X server; however, a value range is more appropriate because of constantly changing system requirements.

6. Again attempt to "startx."

Like before, the SVGA server should fail to start. The error should be apparent from the error messages displayed on the screen (if not then issue the command "startx >& log" to recreate the log file). Once again, examine the log file. The message "(--) SVGA: Mode "800 x 480" needs vert refresh rate of 45.42 Hz. Deleted" indicates that the "VertRefresh" value-range in the "Monitor" section of the XF86Config file should be modified to include the "45.42 Hz" value specified by the SVGA server.

7. Once again, open the XF86Config file for editing.

- Scroll down to the "Monitor" section. Modify the "VertRefresh" value range to include the "45.42" value specified by the SVGA server. To do this, change the lower limit of the value range to a new value below or equal to 45.42 Hz. Set the lower limit to a value such as "45" so that the "VertRefresh" value range reads "45-60." Again, the exact value of 45.42 can be supplied; however, a range is more appropriate because of varying system requirements.

- Save the XF86Config file and exit.

8. Once again issue the "startx" command.

- The SVGA server will again fail to start properly. The message, "(--) SVGA: NM2160: Removing mode (800 x 480) that won't display properly on LCD" appears on screen. The other "(--)" messages can be ignored. The above message informs us that the SVGA server is receiving a message from the NeoMagic card telling it that the "800 x 480" user-defined mode will not display properly on the LCD, so X fails to initialize. This "LCD mode checking" is a form of error checking that is incorporated into the NeoMagic card; however, it is really meant for systems with standard display sizes, not for subnotebooks with "uncommon size panels".<sup>3</sup> Upon reading the "README.neo" file in the "/usr/lib/X11/doc" directory, we found an option that could disable mode checking. The "override\_validate\_mode" option will disable LCD mode checking and should allow for the implementation of the 800 x 480 modeline. There, however, is a warning associated with this option, so proceed carefully.

---

<sup>3</sup> J. Owen and K. Martin, /usr/lib/X11/doc/README.neo ver. 1.1.2.5, Exp., Precision Insight, Inc. (25 June 1999).

"Warning: Disabling mode checking will allow for invalid modes that could damage your LCD."<sup>3</sup>

- Change directories back to the `"/etc/X11"` directory where the XF86Config files are stored.
  - Open the XF86Config file for editing with a text editor such as (Vi, Pico, etc).
  - Scroll down to the "Monitor" section of the XF86Config file. Toward the end of the section, various commented options should be listed that are not used by the X server. This section is where the "override\_validate\_mode" option should be inserted. Insert the line, "Option "override\_validate\_mode"" before the "EndSection" identifier for the "Monitor" section. The override\_validate\_mode option should resolve the mode checking conflicts reported by the NeoMagic card during the initialization of the SVGA server.
  - Save the XF86Config file and exit.
9. Once again issue the "startx" command.
- This time, the SVGA server should be able to start without any problems.
  - If any errors are encountered, log them with "startx >& log" and reexamine the log file. Usually, the errors will be self-explanatory from the error messages given, but sometimes they can be very ambiguous. If X fails to initialize with the settings provided, compare the XF86Config file just created with the one provided in appendix A (designed for the Toshiba Libretto 100CT) that is able to successfully initialize the SVGA X server.

---

<sup>3</sup> J. Owen and K. Martin, [/usr/lib/X11/doc/README.neo](#) ver. 1.1.2.5, Exp., Precision Insight, Inc. (25 June 1999).

---

## 4. Conclusion

---

As illustrated in this document, the definition of a modeline specific to a nonstandard display size is possible. A  $800 \times 480$  modeline for the Toshiba Libretto 100CT was generated with information from both a  $800 \times 600$  modeline and a  $640 \times 480$  modeline. This new modeline allows for maximum screen usage and prevents the display of images off screen. The majority of the technical information presented in this report is based on the tools provided with the Red Hat Linux operating system. Since many other Linux operating systems contain similar tools, this report should serve as a general guide in the development of a modeline specific to a nonstandard display size. Also, the example provided in this report is specific to the problems encountered while developing a working modeline for a Toshiba Libretto 100CT, and therefore the same errors may not occur on other system types. Documentation was a vital source of information when resolving certain hardware or software conflicts. The more you know about what you are trying to use, the better equipped you are in dealing with problems when they arise.

---

## 5. Acknowledgments

---

I would like to acknowledge and thank Brian B. Luu of the U.S. Army Research Laboratory for his mentoring and support in generating this report and for the time he devoted to reviewing and correcting it. Brian is the one who prompted me to the task of finding a way to use the full screen of the Librettos, and then he recommended that I write a report documenting the process. Also, I would like to acknowledge the support of the U.S. Army Research Laboratory, without which this report would not have been possible.

---

## Appendix A. Example of a Working XF86Config file generated for the Toshiba Libretto 100CT

---

```
# XF86Config auto-generated by XF86Setup
#
# Copyright (c) 1996 by The XFree86 Project, Inc.
#
# Permission is hereby granted, free of charge, to any person obtaining a
# copy of this software and associated documentation files (the "Software"),
# to deal in the Software without restriction, including without limitation
# the rights to use, copy, modify, merge, publish, distribute, sublicense,
# and/or sell copies of the Software, and to permit persons to whom the
# Software is furnished to do so, subject to the following conditions:
#
# The above copyright notice and this permission notice shall be included in
# all copies or substantial portions of the Software.
#
# THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY
# KIND, EXPRESS OR
# IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MER-
# CHANTABILITY,
# FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO
# EVENT SHALL
# THE XFREE86 PROJECT BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
# LIABILITY,
# WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF
# OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE
# SOFTWARE.
#
# Except as contained in this notice, the name of the XFree86 Project shall
# not be used in advertising or otherwise to promote the sale, use or other
# dealings in this Software without prior written authorization from the
# XFree86 Project.
#
# See 'man XF86Config' for info on the format of this file

Section "Files"

    RgbPath "/usr/X11R6/lib/X11/rgb"

    FontPath "/usr/X11R6/lib/X11/fonts/misc:unscaled"
    FontPath "/usr/X11R6/lib/X11/fonts/75dpi:unscaled"
    FontPath "/usr/X11R6/lib/X11/fonts/Type1"
```

```

FontPath "/usr/X11R6/lib/X11/fonts/Speedo"
FontPath "/usr/X11R6/lib/X11/fonts/misc"
FontPath "/usr/X11R6/lib/X11/fonts/75dpi"
EndSection
Section "ServerFlags"
EndSection
Section "Keyboard"
    Protocol "Standard"
    XkbRules "xfree86"
    XkbModel "pc101"
    XkbLayout "us"
EndSection
Section "Pointer"
    Protocol "PS/2"
    Device "/dev/mouse"
EndSection
Section "Monitor"
    Identifier "Primary Monitor"
    VendorName "Unknown"
    ModelName "Unknown"
    HorizSync 23.5-35.1
    VertRefresh 45-60
    Modeline "800x480" 25.18 800 840 968 1056 480 491 493 525
EndSection
Section "Device"
    Identifier "Primary Card"
    VendorName "Unknown"
    BoardName "NeoMagic (laptop/notebook)"
    VideoRam 2048
# Chipset "NM2160"
# IOBase 0xfea00000
# MemBase 0xfd000000
# VideoRam 2048
# DacSpeed 90
# Option "linear"
# Option "nonlinear"
# Option "sw_cursor"
# Option "hw_cursor"
# Option "no_accel"
# Option "intern_disp"
# Option "extern_disp"
# Option "mmio"
# Option "no_mmio"
# Option "lcd_center"
# Option "no_stretch"
Option "override_validate_mode"
EndSection

```

```
Section "Screen"
  Driver "SVGA"
  Device "Primary Card"
  Monitor "Primary Monitor"
  DefaultColorDepth 24
  SubSection "Display"
    Depth 8
    Modes "800x480"
  EndSubSection
  SubSection "Display"
    Depth 15
    Modes "800x480"
  EndSubSection
  SubSection "Display"
    Depth 16
    Modes "800x480"
  EndSubSection
  SubSection "Display"
    Depth 24
    Modes "800x480"
  EndSubSection
  SubSection "Display"
    Depth 32
    Modes "800x480"
  EndSubSection
EndSection
```

## Distribution

Admnstr  
Defns Techl Info Ctr  
ATTN DTIC-OCF  
8725 John J Kingman Rd Ste 0944  
FT Belvoir VA 22060-6218

DARPA  
ATTN S Welby  
3701 N Fairfax Dr  
Arlington VA 22203-1714

Ofc of the Secy of Defns  
ATTN ODDRE (R&AT)  
The Pentagon  
Washington DC 20301-3080

Ofc of the Secy of Defns  
ATTN OUSD(A&T)/ODDR&E(R) R J Trew  
3080 Defense Pentagon  
Washington DC 20301-7100

AMCOM MRDEC  
ATTN AMSMI-RD W C McCorkle  
Redstone Arsenal AL 35898-5240

US Army TRADOC  
Battle Lab Integration & Techl Dirctr  
ATTN ATCD-B  
FT Monroe VA 23651-5850

Dir for MANPRINT  
Ofc of the Deputy Chief of Staff for Prsnl  
ATTN J Hiller  
The Pentagon Rm 2C733  
Washington DC 20301-0300

SMC/CZA  
2435 Vela Way Ste 1613  
El Segundo CA 90245-5500

US Army ARDEC  
ATTN AMSTA-AR-TD  
Bldg 1  
Picatinny Arsenal NJ 07806-5000

US Army Info Sys Engrg Cmnd  
ATTN AMSEL-IE-TD F Jenia  
FT Huachuca AZ 85613-5300

US Army Natick RDEC Acting Techl Dir  
ATTN SBCN-T P Brandler  
Natick MA 01760-5002

US Army Simulation Train & Instrmntn  
Cmnd  
ATTN AMSTI-CG M Macedonia  
ATTN J Stahl  
12350 Research Parkway  
Orlando FL 32826-3726

US Army Tank-Automtv Cmnd RDEC  
ATTN AMSTA-TR J Chapin  
Warren MI 48397-5000

Nav Surfc Warfare Ctr  
ATTN Code B07 J Pennella  
17320 Dahlgren Rd Bldg 1470 Rm 1101  
Dahlgren VA 22448-5100

Hicks & Assoc Inc  
ATTN G Singley III  
1710 Goodrich Dr Ste 1300  
McLean VA 22102

US Army Rsrch Lab  
ATTN AMSRL-CI N Radhakrishnan  
Aberdeen Proving Ground MD 21005-5067

Director  
US Army Rsrch Lab  
ATTN AMSRL-RO-D JCI Chang  
PO Box 12211  
Research Triangle Park NC 27709



## Distribution (cont'd)

US Army Rsrch Lab

ATTN AMSRL-CI J D Gantt

ATTN AMSRL-CI-AI-R Mail & Records

Mgmt

ATTN AMSRL-CI-AP Techl Pub

ATTN AMSRL-CI-C J Gowens

ATTN AMSRL-CI-CN G Cirincione

ATTN AMSRL-CI-CN H Harrelson

ATTN AMSRL-CI-LL Techl Lib (2 copies)

ATTN AMSRL-CI-SD R Gopaul (10 copies)

REPORT DOCUMENTATION PAGE			Form Approved OMB No. 0704-0188	
Public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Washington Headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302, and to the Office of Management and Budget, Paperwork Reduction Project (0704-0188), Washington, DC 20503.				
1. AGENCY USE ONLY (Leave blank)		2. REPORT DATE April 2001		3. REPORT TYPE AND DATES COVERED Final, FY 2000
4. TITLE AND SUBTITLE Creation of a New Modeline for a Nonstandard, Unsupported Display Size in a Red Hat Linux, XFree86 Environment			5. FUNDING NUMBERS DA PR: AH48 PE: 61102A	
6. AUTHOR(S) Richard D. Gopaul				
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) U.S. Army Research Laboratory Attn: AMSRL-CI-SD email: RGopaul@arl.army.mil 2800 Powder Mill Road Adelphi, MD 20783-1197			8. PERFORMING ORGANIZATION REPORT NUMBER ARL-TR-853	
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES) U.S. Army Research Laboratory 2800 Powder Mill Road Adelphi, MD 20783-1197			10. SPONSORING/MONITORING AGENCY REPORT NUMBER	
11. SUPPLEMENTARY NOTES ARL PR: 1FEPA1 AMS code: 611102H4811				
12a. DISTRIBUTION/AVAILABILITY STATEMENT Approved for public release; distribution unlimited.			12b. DISTRIBUTION CODE	
13. ABSTRACT (Maximum 200 words) Nonstandard display modes for irregular display sizes are not readily supported by the Red Hat Linux operating system. This lack of support leads to the wasting of screen space when one uses either the text mode or X Windows interfaces. The problem becomes especially prevalent in the X environment. The system will not correctly detect the size of the screen and may project images off screen or images that are too small to take advantage of the full viewable capacity of the screen. To correct this problem, a new modeline must be created that defines the total viewable area of the monitor. This newly created modeline will allow the X server to display any and all images within the bounds of the screen. To accomplish this task, one must modify the XF86Config file to include the newly created modeline, as well as, any display options necessary to the display adapter for proper image display. This report focuses specifically on the creation of a nonstandard "800 x 480" modeline for the Toshiba Libretto 100CT. This report should serve as an adequate guide for the creation of other nonstandard modelines.				
14. SUBJECT TERMS XFree 86, Modeline, Libretto			15. NUMBER OF PAGES 26	
			16. PRICE CODE	
17. SECURITY CLASSIFICATION OF REPORT Unclassified	18. SECURITY CLASSIFICATION OF THIS PAGE Unclassified	19. SECURITY CLASSIFICATION OF ABSTRACT Unclassified	20. LIMITATION OF ABSTRACT UL	